Johnathan E. Mansfield, OSB No. 055390
john@mansfieldlaw.net
MansfieldLaw
11 SW Morrison, Suite 400
Portland, OR 97204
Phone: 971.271.8615

Marc A. Fenster (pro hac vice)
mfenster@raklaw.com
Andrew D. Weiss (pro hac vice)
aweiss@raklaw.com
Russ August & Kabat
12424 Wilshire Blvd., 12th Floor
Los Angeles, CA 90025
Telephone 310.826.7474
Fax 310.826.6991

Attorneys for Plaintiff
Tranxition, Inc.

<div align="center">

IN THE UNITED STATES DISTRICT COURT
FOR THE DISTRICT OF OREGON
PORTLAND DIVISION

</div>

TRANXITION, INC., **a Delaware corporation,**	Civil Case No. 3:12-cv-1403-BR
Plaintiff, vs.	**PLAINTIFF TRANXITION, INC.'S OPENING CLAIM CONSTRUCTION BRIEF**
APPSENSE, INC., **a Delaware corporation,**	
Defendant.	

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF AUTHORITIES

PAGE

CASES

TABLE OF AUTHORITIES

PAGE

iv

I.   INTRODUCTION.

The Court should adopt Tranxition's proposed claim constructions because they are based on the intrinsic evidence and accurately apply well-accepted claim construction principals. The claims of the '170 patent are written in straightforward language and a person of ordinary skill in the art understands the meaning of the claim language.

AppSense's proposed constructions, on the other hand, repeatedly import limitations that are not supported by the intrinsic evidence and claim construction principals. For example, for the "executes before other code of the process," AppSense proposes that this straightforward language be construed using the same words except adding the word "any": "executes before *any* other code of the process." Although AppSense has only disclosed citations, without explanation, none of the evidentiary citations clearly and unambiguously import the "any" limitation into this claim term. Without clear and unambiguous disclaimer, claim construction doctrine does not support importing limitations into claim language as proposed by AppSense. Similarly, AppSense argues that a subset of the claim limitations that use "component" should be deemed to be means-plus-function limitations and construed in the limited manner required by means-plus-function doctrines. Both the law and facts contradict AppSense's position. Courts that have addressed arguments very similar to those that will presumably be made by AppSense have been repeatedly rejected. And, similar to the relevant case law, the specification of the '170 patent provides extensive discussion and detail regarding the components being claims. The "component" terms are not means-plus-function terms.

Accordingly, as will be discussed in more detail below, the Court should adopt Tranxition's proposed constructions.

II.   AGREED CONSTRUCTIONS.

The parties have agreed on the constructions for the following terms:

Term	Agreed Construction
Personality information	Collection of configuration settings and files that represent the personality of the computer or other device

Configuration information	Information related to configuration settings
Personality information store	Computer memory that stores personality information
Persistent storage medium	Disk drives or other persistent storage

Additionally, AppSense initially suggested that the following terms needed construction but has since agreed that the following terms do not need to be construed and are therefore subject to their plan and ordinary meaning:

- Personality

- Adopting a personality of a source

- Software that is automatically executed when the storage medium is connected to the source or the target

- Application

- Application that is to execute as the process

- Function

- Personality information of the source is not stored on to a personality information store of the target

- Personality information extracted from the source

- Functions that are to be intercepted[1]

- Intercept the invocation of the located function[1]

- Modifies information of the [application/process] to effect the interception of request[1]

III.   THE LAW OF CLAIM CONSTRUCTION.

---

[1] AppSense initially proposed that these three terms was indefinite. On the day before AppSense was due to provide the bases for AppSense's position, AppSense withdrew its proposed construction and instead agreed that no construction was necessary.

A.   **Claims Must Be Construed According To The Intrinsic Evidence.**

"It is a 'bedrock principle' of patent law that 'the claims of a patent define the invention to which the patentee is entitled the right to exclude.'" *Phillips v. AWH Corp.*, 415 F.3d 1303, 1312 (Fed. Cir. 2005) (quoting *Innova/Pure Water, Inc. v. Safari Water Filtration Sys., Inc.*, 381 F.3d 1111, 1115 (Fed. Cir. 2004)); *see also Vitronics Corp. v. Conceptronic, Inc.*, 90 F.3d 1576, 1582 (Fed. Cir. 1996) ("we look to the words of the claims themselves . . . to define the scope of the patented invention"). Thus, a disputed claim term must be considered in the context of the entire claim. *Phillips*, 415 F.3d at 1314 ("To begin with, the context in which a term is used in the asserted claim can be highly instructive.").

"It is well settled that, in interpreting an asserted claim, the court should look first to the intrinsic evidence of record, i.e. the patent itself, including the claims, the specification, and, if in evidence, the prosecution history." *Vitronics*, 90 F.3d at 1582. "In most situations, an analysis of the intrinsic evidence alone will resolve any ambiguity in a disputed claim term. In such circumstances, it is improper to rely on extrinsic evidence." *Id.* at 1583. If the Court does choose to rely on extrinsic evidence, the Court must be sure to consider the flaws inherent in such evidence while weighing its effect on the intrinsic evidence. *Phillips*, 415 F.3d at 1318-19 (describing the flaws of relying on extrinsic evidence); *see also Altiris, Inc. v. Symantec Corp.*, 318 F.3d 1363, 1369 (Fed. Cir. 2003) ("Extrinsic evidence may never be relied upon, however, to vary or contradict the clear meaning of terms in the claims.").

Accordingly, the Court should rely on the intrinsic evidence to construe the disputed claim terms and can only rely on extrinsic evidence if the term cannot be construed using the intrinsic evidence alone. To the extent that extrinsic evidence varies or changes the intrinsic evidence, it should be disregarded.

B.   **Claim Terms Are Given Their Ordinary And Customary Meaning
Except Where The Patentee Clearly And Unmistakably Intended A
Different Meaning.**

Claim terms should be construed in accordance with their ordinary and customary meaning, as a person of ordinary skill in the art would understand them at the time of the

invention. *Phillips*, 415 F.3d at 1312-13 ("We have frequently stated that the words of a claim 'are generally given their ordinary and customary meaning.' We have made clear, moreover, that the ordinary and customary meaning of a claim term is the meaning that the term would have to a person of ordinary skill in the art in question at the time of the invention, i.e., as of the effective filing date of the patent application." (quoting *Vitronics*, 90 F.3d at 1582) (internal citations removed)). Indeed, there is a "heavy presumption" that claim terms will be given their ordinary and customary meaning. *CCS Fitness, Inc. v. Brunswick Corp.*, 288 F.3d 1359, 1366 (Fed. Cir. 2002), *cert. denied*, 123 S. Ct. 2230 (2003).

Although claim terms are generally construed according to their ordinary meaning, the law permits a patentee to specially change their meaning if, and only if, it is clear from the intrinsic record that the patentee intended to do so. *Vitronics*, 90 F.3d at 1582 ("Although words in a claim are generally given their ordinary and customary meaning, a patentee may choose to be his own lexicographer and use terms in a manner other than their ordinary meaning, as long as the special definition of the term is clearly stated in the patent specification or file history."); *Teleflex, Inc. v. Ficosa N. Am. Corp.*, 299 F.3d 1313, 1324 (Fed. Cir. 2002) ("[W]e conclude that claim terms take on their ordinary and accustomed meanings unless the patentee demonstrated an intent to deviate from the ordinary and accustomed meaning of a claim term by redefining the term or by characterizing the invention in the intrinsic record using words or expressions of manifest exclusion or restriction, representing a clear disavowal of claim scope."). The disavowal of claim scope must be so clear and unambiguous that the public could rely on the representation. *SanDisk Corp. v. Memorex Prods., Inc.*, 415 F.3d 1278, 1287 (Fed. Cir. 2005) ("An ambiguous disclaimer, however, does not advance the patent's notice feature or justify public reliance, and the court will not use it to limit a claim term's ordinary meaning. There is no 'clear and unmistakable' disclaimer if a prosecution argument is subject to more than one reasonable interpretation, one of which is consistent with a proffered meaning of the disputed term." (internal citations removed)); *Omega Eng'g, Inc. v. Raytek Corp.*, 334 F.3d 1314, 1325-26

4

(Fed. Cir. 2003) ("Consequently, for prosecution disclaimer to attach, our precedent requires that the alleged disavowing actions or statements made during prosecution be both clear and unmistakable.").

Thus, absent a clear disclaimer of the scope of the claims, the disputed claim terms should be given their common, ordinary meanings.

C.   Limitations From the Specification Must Not Be Imported Into The Claims.

In claim construction, the danger of reading limitations from the specific embodiments described in the specification must be avoided. *Phillips*, 415 F.3d at 1319-1320 ("one of the cardinal sins of patent law [is] reading a limitation from the written description into the claims." (quoting *SciMed Life Sys., Inc. v. Advanced Cardiovascular Sys., Inc.*, 242 F.3d 1337, 1340 (Fed. Cir. 2001)); *Playtex Prods., Inc. v. Procter & Gamble Co.*, 400 F.3d 901, 907 (Fed. Cir. 2005) ("By its reliance on the figures, the district court improperly limited claim 1 to a preferred embodiment.   We have consistently advised against this approach to claim construction."); *nCube Corp. v. SeaChange Int'l, Inc.*, 436 F.3d 1317, 1321-22 (Fed. Cir. 2006) (affirming a judgment of willful infringement, the Federal Circuit recognized the claims intended a broader scope than the described embodiments and upheld the finding that a patent on a method and apparatus for multimedia data networking was not limited to the specification's mention of routing data messages using logical addresses); *see also Johnson Worldwide Assocs., Inc. v. Zebco Corp.*, 175 F.3d 985, 989 (Fed. Cir. 1999) ("General descriptive terms will ordinarily be given their full meaning; modifiers will not be added to broad terms standing alone."). "Claims of a patent may only be limited to a preferred embodiment by the express declaration of the patentee." *Playtex*, 400 F.3d at 908; *see also Rexnord Corp. v. Laitram Corp.*, 274 F.3d 1336, 1344-45 (Fed. Cir. 2001) ("Our case law is clear that an applicant is not required to describe in the specification every conceivable and possible future embodiment of his invention."). "Even when the specification describes only a single embodiment, the claims of the patent will not be read restrictively unless the patentee has demonstrated a clear intention to limit the claim scope

5

using 'words or expressions of manifest exclusion or restriction." *Liebel-Flarsheim Co. v. Medrad, Inc.*, 358 F.3d 898, 906 (Fed. Cir. 2004) (quoting *Teleflex,* 299 F.3d at 1327) (reversing district court's holding that injector claims were limited to injectors with pressure jackets, even though every embodiment in the specification had pressure jackets).

Therefore, without an express indication to the contrary, limitations from the specification should not be imported into the claims.

> **D.   Where The Plain And Ordinary Meaning Applies, A Court Need Not Expressly Construe A Term.**

When the plain and ordinary meaning of a term is the appropriate construction, the Court need not provide a construction simply because one party has suggested that a term needs to be construed. Instead, a Court need only construe a term where the parties have "a fundamental dispute regarding the scope of a claim. *O2 Micro Int'l Ltd. v. Beyond Innovation Tech. Co.*, 521 F.3d 1351, 1362 (Fed. Cir. 2008)). Thus, in situations for example where the dispute is only whether certain additional language should be imported into existing claim language, a Court may reject the imported language and simply hold that the plain and ordinary meaning of the term applies. *See, e.g., id.*; *Finjan, Inc. v. Secure Computing Corp.*, 626 F.3d 1197, 1207 (Fed. Cir. 2010) (upholding district court's rejection of defendant's proposed construction and finding no construction necessary).

IV.   TRANXITION'S PROPOSED CONSTRUCTIONS ARE BASED UPON PROPER CLAIM CONSTRUCTION PRINCIPLES AND SHOULD BE ADOPTED.

The parties currently dispute the construction of 16 terms of the '170 patent. Tranxition's proposed constructions should be adopted because they are consistent with the plain and ordinary meaning of the disputed terms in light of the intrinsic evidence. AppSense's proposed constructions, on the other hand, seek to improperly import limitations that are inconsistent with the plain meaning of the term and the related intrinsic evidence.

Each disputed term is discussed individually below. In spite of the Court's stated preference and Order (D.I. 48), aside from expert opinion, the only bases for its claim

constructions provided by AppSense were evidentiary citations to the '170 patent and its file history. *See* Declaration of Andrew D. Weiss In Support Of Plaintiff Tranxition, Inc.'s Opening Claim Construction Brief ("Weiss Decl.") at Ex. A. Tranxition therefore rebuts the arguments that Tranxition believes AppSense may bring based on the provided evidentiary citations. Tranxition reserves the right to supplement this Brief or provide supplemental or amended constructions or arguments at the claim construction hearing in light of AppSense's arguments.

A. Target[2]: "Computing Device That Adopts The Personality Stored On The Storage Medium."

Tranxition's Construction	AppSense's Construction
Computing device that adopts the personality stored on the storage medium	A computing device separate from the source that adopts the personality information of the source

Tranxition's proposed construction for target should be adopted because it is consistent with the plain meaning of the term in light of the intrinsic evidence. Target is used according to its plain meaning in the claims. *E.g.*, '170 patent at claim 1 ("intercepting requests to retrieve personality information of the target"). The specification similarly uses target consistent with its plain and ordinary meaning. For example, the specification expressly discloses: "The source and target may be computing devices such as a personal computer, personal digital assistant, cell phone, digital camera, other consumer electronic device, and so on." '170 patent at 2:24-27. The specification further discloses that the target is the computing device that adopts the stored personality through the use of interception. *E.g.*, '170 patent at 2:28-34 ("The personality system intercepts requests of the target to retrieve personality information….   When a request is intercepted, the personality system retrieves the personality information from the connected storage system, rather than from the personality information of the target."); 4:41-48 ("When the user specifies to use the device as a target, the personality system then loads the interceptor component onto the target as an interceptor component 121. The interceptor component then

---

[2] "Target" can be found in asserted claims 1, 5-8, 10-12, 15, 19 and 22-23 of the '170 patent.

installs code to intercept requests for personality information and to redirect the requests to the interceptor component so that the requests can be serviced from the personality information of the storage medium."). The file history does not alter the ordinary meaning of this term. Thus, Tranxition's proposed construction should be adopted.

AppSense's proposed construction, on the other hand, should be rejected because it imports the limitation that the target be "separate from the source." The plain meaning of the term, as discussed above, does not require that the target device be different from the target device. In fact, contrary to AppSense's proposed construction, the specification expressly discloses an embodiment where the device adopting the personality (*i.e.*, the target) is the same as the source:

> When the storage medium is connected to a device, such as a source 110, the auto execute component automatically executes and prompts the user to indicate whether to extract the personality of the source *or to have the source adopt a personality based on personality information stored on the storage medium*."

'170 patent at 3:61-66 (emphasis added). A person of ordinary skill in the art understands that, consistent with the claim language, a source is acting as a target (as used by the claims) when it adopts a personality. AppSense has not pointed to any clear and unambiguous disclaimers of the full scope of the target term. Therefore, AppSense's imported limitation requiring that the source and target be different devices is inappropriate and should be rejected. *See, e.g.*, *SanDisk*, 415 F.3d at 1287.

Accordingly, the Court should construe target as proposed by Tranxition: "computing device that adopts the personality stored on the storage medium."

B. Source:[3] "Computer Device From Which the Personality Information Is Extracted."

Tranxition's Construction	AppSense's Construction
Computing device from which the personality information is extracted	A computing device separate from the target that stores its personality information on a storage medium

---

[3] "Source" can be found in asserted claims 1, 10-12, 15, 17 and 22-23 of the '170 patent.

Source is the companion term to the target term. Where as a target adopts a personality, the source is the device from which the personality is derived. *E.g.*, '170 patent at 2:61-3:3, 3:61-66. For the same reasons discussed above, Tranxition's proposed construction adopts the plain meaning of source and nothing in the claims, specification and file history alters the plain meaning of the term source.

AppSense's proposed construction introduces two limitations that are improper. First, like target, AppSense proposes that the source and target must be different devices. For the same reasons discussed with respect to target, AppSense's proposed limitation must be rejected. *See, e.g.*, '170 patent at 3:61-66. Second, AppSense proposes to import the functional limitation that the source "stores its personality information on a storage medium." The plain meaning of source does not require that it store personality on a storage medium. Further, the intrinsic evidence does not support this limitation. Consistent with Tranxition's proposed construction, the claims require that the personality be extracted from the source. The claims do not require, however, that the source actually perform the extraction. *E.g.*, '170 patent at claim 17 ("personality information extracted from the source"). Similarly, the specification contains no clear and unambiguous disclaimer requiring that the source itself store the personality information on a storage medium. Instead, in one preferred embodiment, an extraction component, and not the source, stores the extracted personality on a storage medium. *E.g.*, '170 patent at 3:2-3, 4:30-33. While the extraction component may be a process running on the source, a person of ordinary skill in the art understands that the extraction component could run on a device other than a source, for example a server with access to the source. Because AppSense cannot cite a clear and unambiguous disclaimer importing AppSense's proposed limitation, AppSense's proposed construction should be rejected. *See, e.g.*, *SanDisk*, 415 F.3d at 1287.

Accordingly, the Court should construe source as proposed by Tranxition: "computing device from which the personality information is extracted."

C. Storage Medium:[4] "Computer Memory."

Tranxition's Construction	AppSense's Construction
Computer memory	A device that can be connected to the source or target for storing personality information of the source

The Court should adopt Tranxition's proposed construction because it is consistent with the plain meaning of storage medium. The claims use storage medium simply to refer to a computer memory where information is stored. *E.g.*, '170 patent at claim 1 ("providing a storage medium containing personality information of the source"). The specification similarly describes storage medium in its ordinary sense. *E.g.*, '170 patent at 2:21-22 (describing an exemplary embodiment of a removable storage memory as a USB device); 5:43-45 (describing an exemplary embodiment of a storage medium as part of a server); 6:11-17 (describing storage media, a similar term, as including disk drives, USB devices and removable disk storage, among other types of memory). The file history also does not alter the plain meaning of storage medium. Thus, consistent with the plain and ordinary meaning of storage medium, storage medium should be construed broadly as "storage memory."

AppSense's proposed construction imports limitations into the plain meaning. First, AppSense proposes to import the limitation that a storage medium is a "device that can be connected;" in other words, AppSense proposes to limit a storage medium to a device such as a USB drive. This limitation should be rejected because it improperly imports limitations from preferred embodiments described in the specification, which is contrary to claim construction doctrine.[5] *E.g.*, *Phillips*, 415 F.3d at 1319-1320. Additionally, the specification expressly

---

[4] "Storage medium" can be found in asserted claims 1, 4, 7-8, 10-12 and 17 of the '170 patent.

[5] Additionally, when the claims meant to draw a distinction between a removable and persistent storage memory, the claim expressly do so. *See* '170 patent at claim 8.

describes embodiments where the storage medium is used generally and is not limited to a removable storage medium. '170 patent at 5:43-45 ("In one embodiment, the storage medium may be part of a server that is connected to source and targets via a network, such as the Internet."). This limitation should therefore be rejected.

Second, AppSense proposes to import the functional limitation that a storage medium needs to store the personality information of the source. In addition to being inconsistent with the plain and ordinary meaning of the term, AppSense's proposed construction, conflicts with the claim language. Claim 8 of the '170 patent requires that the retrieved personality information of the source not be stored by the persistent *storage medium* of the target. If the proper construction of storage medium requires storing personality information of the source, as proposed by AppSense, how can the storage medium of the target in claim 8 not store personality information of the source? This conflict cannot be resolved. Because the second proposed limitation is inconsistent with the plain and ordinary meaning of storage medium as well as its use in the claims, AppSense's second proposed limitation must also be rejected.

The proper construction of storage memory is therefore "computer memory.

D.   Intercept/Interception/Intercepting Requests/Of Requests/The Requests:[6] No Construction Necessary.

Tranxition's Construction	AppSense's Construction
Plain and ordinary meaning	Intercepting requests from a process

Although this term as proposed by AppSense is presented in a confusing manner, it appears that AppSense is seeking to construe all instances of claim language requiring that requests be intercepted, such as "intercepting requests" and "intercepting of requests" in claim 1 and "intercepted" in claim 14. The intrinsic evidence does not alter the plain and ordinary meaning of the various "intercepting" terms. *E.g.*, '170 patent at 4:41-5:8. AppSense appears to

---

[6] According to AppSense, these terms are found in asserted claims 1, 6, 11-12, 14-15, 20 and 22-23 of the '170 patent.

agree by using the language of "intercepting requests" in its proposed construction. Thus, the Court need not construe these claims terms.

AppSense appears to agree that the "intercept" language used by the claims is clear. AppSense proposes a construction of these terms, however, to import the limitation that the intercepted requests must be "from a process." As an initial matter, AppSense's proposed construction is nonsensical for some of the terms being construed. For example, in claim 14, the claim language is "modifying references to functions that are to be intercepted." Replacing "intercepted" with AppSense's proposed construction ("intercepting requests from a process") makes claim 14 nonsensical. AppSense's proposed "from a process" limitation should also be rejected because there is no clear and unambiguous disclaimer importing this limitation into the claims. In fact, none of the evidence cited by AppSense ('170 patent at Fig. 5, 3:7-27, 4:64-5:8) even uses the term "process" nor does the specification import the limitation that the interception can only be of requests from a process.[7] AppSense's proposed construction should therefore be rejected.

Accordingly, the Court should reject AppSense's proposed imported limitation and should instead decline to construe this term. *See, e.g.*, *O2 Micro*, 521 F.3d 1362.

E.   Loaded As Part Of Each Process:[8] No Construction Necessary.

Tranxition's Construction	AppSense's Construction
Plain and ordinary meaning	Loaded **by the operating system** as part of each process **that it creates**

---

[7] AppSense's proposed construction is also inappropriate because, under AppSense's own theory, it would render claim 12 indefinite. AppSense (and its expert) argue that claim 13 is indefinite because it refers to modifying the code of the process, while the independent claim 12 refers to modifying the code of the application. *See infra*. AppSense's proposed construction also introduces the process concept into claim 12. Under AppSense's theory, therefore, claim 12 would likely be indefinite if its proposed construction is adopted. The Court should not import limitations that render claims invalid.

[8] "Loaded As Part Of Each Process" can be found in asserted claims 1, 11-12, 15 and 22-23 of the '170 patent.

Because this term is written in clear, understandable language, this term needs no construction. The intrinsic evidence does not alter the plain meaning of this term. AppSense proposes to construe the term using the existing claim language but AppSense also proposes to import two additional limitations, reflected in bold in the table above. AppSense first proposes that the loading must be done "by the operating system." The claims do not have this limitation. The specification also contains no such limitation. While the specification does disclose that operating system "may" load an interceptor dynamic link library ('170 patent at 4:48-50), other preferred embodiments simply refer to loading the startup interceptor DLL, without requiring that a particular component perform the loading (*e.g.*, '170 patent at 7:47-49). There is therefore no clear and unambiguous disclaimer importing AppSense's proposed limitation. The imported limitation should therefore be rejected. *See, e.g.*, *SanDisk*, 415 F.3d at 1287.

AppSense's second imported limitation is "that it creates." For the same reasons as the first proposed limitation, the intrinsic evidence does not contain any clear and ambiguous disclaimers importing this limitation into this term. Additionally, AppSense's proposed limitation should be rejected because it is ambiguous. Does "it" refer to the operating system or the process or something else?

Accordingly, the Court should reject AppSense's proposed imported limitations and should instead hold that this term needs no construction. *See, e.g.*, *O2 Micro*, 521 F.3d 1362.

F.   Executes Before Other Code Of The Process/Executed Before Other Code Of The Process:[9] No Construction Necessary.

Tranxition's Construction	AppSense's Construction
Plain and ordinary meaning	Executes before **any** other code of the process

Like the terms above, AppSense appears to agree that the existing claim language is clear and need no construction. AppSense asserts, however, that the intrinsic evidence requires

---

[9] These terms can be found in asserted claims 1, 11-12, 15 and 22-23 of the '170 patent.

importing the "any" limitation into these terms, as shown in bolded text in the table above. There is no clear and ambiguous disclaimer importing this limitation into the claim language and it should therefore be rejected. *See, e.g., SanDisk*, 415 F.3d at 1287. AppSense's proposed limitation appears to be based on the single use of "any" in the specification. '170 patent at 4:48-53. AppSense's cited portion of the specification, however, expressly discloses it is an example and is not limiting. *Id.* Further, additional embodiments using the same claim language without the imported "any" limitation. *E.g.*, '170 patent at 7:29-32. Limitations should not be imported from preferred embodiments described in the specification. *See, e.g., Playtex*, 400 F.3d at 907. Accordingly, the Court should reject AppSense's proposed imported limitation and should instead find that these terms need no construction. *See, e.g., O2 Micro*, 521 F.3d 1362.

G. Setting A Registry Entry Of A System Registry/Wherein A Registry Entry Of A System Registry Is Set:[10] No Construction Necessary

Tranxition's Construction	AppSense's Construction
Plain and ordinary meaning	Setting**, by the install DLL, the value of** a registry entry of a system registry

Like the previous terms, AppSense adopts the original language of these terms and simply proposes to import additional limitations not found in the claim term. As shown in bold in the table above, AppSense proposes to import the limitations that the registry entry is set by an install DLL and the setting involves setting a value of a registry entry. AppSense cannot, however, point to any clear and unmistakable disclaimers importing these limitations into the claim language. *See, e.g., SanDisk*, 415 F.3d at 1287. Instead, AppSense points only to Figure 6 and its accompanying description in the specification as support for its proposed construction ('170 patent at 7:28-45). Figure 6, however, expressly describes only one embodiment ('170 patent at 2:7-8) and the specification describing Figure 6 also emphasizes that it is describing one embodiment ('170 patent at 7:29). AppSense also ignores other embodiments that involve

---

[10] These terms can be found in asserted claims 1 and 15 of the '170 patent.

setting a registry entry without the express limitation that an install DLL performs the setting. *E.g.*, '170 patent at 3:9-19 (describing the use of an adoption component). The claim language does not limit itself to a particular embodiment and, without clear and unambiguous disclaimers, it is improper to import such limitations in claim construction. *See, e.g., Playtex*, 400 F.3d at 907. The Court should therefore reject AppSense's proposed construction and the Court should hold that this term needs no construction. *See, e.g., O2 Micro*, 521 F.3d 1362.

H.   Startup Interceptor DLL:[11] "Dynamic Link Library That Is Loaded As Part Of A Process That Executed Before Other Code Of The Process."

Tranxition's Construction	AppSense's Construction
Dynamic link library that is used at least in part in the interception of requests that is loaded as part of a process and that executes before other code of the process	A dynamic link library that identifies the application that is to execute as the process and locates the function to be intercepted and installs the code to intercept the invocation of the function

The parties agree that the startup interceptor DLL is a dynamic link library. *See*, '170 patent at 4:48-53. The dispute is the remainder of the proper construction of this term. Tranxition's proposed construction should be adopted because it is consistent with the intrinsic evidence. The claims use startup interceptor DLL in its ordinary manner. For example, claim 1 requires that the startup interceptor DLL be involved in the intercepting of requests and that the DLL be loaded as part of a process and execute before other code of the process. Claim 15 includes similar requirements for the startup interceptor DLL but also includes the narrower requirement that the startup interceptor DLL be the component that intercepts the requests. Like the claims, in the specification, the startup interceptor DLL is described as being involved in the interception functionality ('170 patent at Fig. 7, 7:46-8:3) and as being a DLL that is loaded as part of a created process and before other code of the process ('170 patent at 7:30-32). The file

---

[11] "Startup interceptor DLL" can be found in asserted claims 1, 11-12, 15 and 22-23 of the '170 patent.

history does not alter the meaning of startup interceptor DLL as provided in the specification. Thus, Tranxition's proposed construction reflects the ordinary meaning of startup interceptor DLL in light of the intrinsic evidence.

AppSense's proposed construction should be rejected because it improperly imports a preferred embodiment into the term. *See, e.g., Playtex*, 400 F.3d at 907. Tranxition believes AppSense will rely on Figure 7 and its accompanying text ('170 patent at 7:46-8:3) as support for its proposed construction.[12] AppSense's cited evidence, however, expressly discloses that it is only describing "one embodiment." '170 patent at 7:46-47, 2:9-10. The specification further acknowledges that there are multiple known ways to perform interception. '170 patent at 4:64-5:8. The specification does not limit the capabilities of the startup interceptor DLL to a single particular functionality. The improper nature of AppSense's proposed construction is further supported by the conflict between AppSense's proposed construction and the claim language. In AppSense's proposed construction, the startup interceptor DLL "installs the code to intercept the invocation of the function." Claim 15, however, requires that the startup interceptor DLL be the component that intercepts the requests. It is contradictory to require the startup interceptor DLL to install code to intercept requests while at the same time requiring that the startup interceptor DLL actually perform the interception functionality.

Accordingly, Tranxition's proposed construction is consistent with the intrinsic and should be adopted.

I. Process:[13] "An Application That Is Executed On A Target Device."

Tranxition's Construction	AppSense's Construction
An application that is being executed on a target device	The execution of an application on a target device

---

[12] AppSense also cites the file history of the '170 patent but the file history does not contain any statements about the meaning of startup interceptor DLL. Because the file history does not contain clear and unambiguous disclaimers of the ordinary meaning of startup interceptor DLL, It is unclear why AppSense believes the file history supports its proposed construction.

[13] "Process" can be found in asserted claims 1, 6, 11-13, 15, 20 and 22-24 of the '170 patent.

The parties agree as to the general meaning of this term. The dispute, however, has to do with the reduction of the meaning to words. The Court should adopt Tranxition's proposed construction. Process is used as a noun in the claims. *E.g.*, '170 patent at claim 1 ("a startup interceptor DLL that is to be loaded as part of each **process** and that executes before other code of the **process**"). Tranxition's proposed construction reflects the use of this word in the claims. AppSense's proposed construction, however, defines process as if it were used as a verb in the claims. The claims do not use process as a verb. Accordingly, the Court should adopt Tranxition's proposed construction.

J.   **Canonical Form:[14] No Construction Necessary But If A Construction Is Necessary, "A Standard Form."**

Tranxition's Construction	AppSense's Construction
Plain and ordinary meaning If a construction is necessary: "a standard form"	A neutral form of personality information that is independent of any application program or the operating system under which an application program executes

The Court should adopt Tranxition's proposed construction because it is consistent with the plain and ordinary meaning of canonical form. The claims use the term in an ordinary manner. The specification further discloses that a canonical form "may be a neutral form or may default to the form specific to an application or an operating system." '170 patent at 3:53-55. AppSense's proposed construction should be rejected because it eliminates a specifically disclosed preferred embodiment where a canonical form can use a "default" form from a specific application or operating system. A claim construction that excludes a preferred embodiment is not appropriate. *See, e.g., Playtex*, 400 F.3d at 907. Tranxition expects that AppSense will argue that its construction is appropriate because it adopts most of the words used at 3:30-33 of the '170 patent. AppSense's argument, however, takes the disclosure of the specification out of

---

[14] This term can be found in asserted claims 4-5 and 18-19 of the '170 patent.

context. The reference to a canonical form being independent of an application or operating system was used to give a "sense" of the neutral nature of the canonical form. '170 patent at 3:30-33. It was not meant to exclude a canonical form that is a form used by a particular application or operating system, as the remainder of the paragraph at 3:28-55 of the '170 patent describes. Accordingly, the Court should adopt Tranxition's proposed construction.

K. Target-Specific Form:[15] No Construction Necessary But If A Construction Is Necessary: "Form Of Personality Information Specific To A Target."

Tranxition's Construction	AppSense's Construction
Plain and ordinary meaning If a construction is necessary: "form of personality information specific to a target"	A form of personality information specific to **the applications** of the target

This term needs no construction. The intrinsic evidence uses this term consistent with its plain and ordinary meaning as a form specific to a target. *E.g.*, '170 patent at claim 5 ("converting the retrieved information from the canonical form to a target-specific form"), 3:46-48 ("converts the personality information to a target-specific form when access by the target"). The plain meaning of the term is emphasized by the construction proposed by AppSense that is essentially a rewording of the term. As shown in bold, AppSense proposes to import the additional limitation that the form must be specific to "the applications" of the target. AppSense asserts that 3:38-42 of the '170 patent support its construction. AppSense ignores, however, the very next sentences that provides additional examples that use target-specific form to refer to personality information for an operating system as well as an application. '170 patent at 3:42-53. AppSense's proposed limitation must therefore be rejected because it improperly imports limitations from a preferred embodiment and excludes other preferred embodiments. *See, e.g.*, *Playtex*, 400 F.3d at 907. If a construction is necessary, the Court should enter the construction proposed by Tranxition, which is consistent with the plain meaning.

---

[15] This term can be found in asserted claims 5, 19 of the '170 patent.

L.   AppSense's Indefiniteness Arguments Should Be Rejected Because AppSense Has Not Shown By Clear And Convincing Evidence That Any Terms Of The Asserted Claims Of The '170 Patent Are Indefinite.

The definiteness analysis requires a determination of "whether one skilled in the art would understand the bounds of the claim when read in light of the specification." *Miles Lab. v. Shandon, Inc.*, 997 F.2d 870, 875 (Fed. Cir. 1993). A claim is definite as long as it is not "insolubly ambiguous," even if the task of construing the term is "formidable." *See, e.g., Exxon Research & Eng'g Co. v. U.S.*, 265 F.3d 1371, 1375 (Fed. Cir. 2001). A claim is also definite even if reasonable persons disagree regarding the construction. *Id.* Defendants bear the heavy burden of showing that a term is indefinite by clear and convincing evidence. *See, e.g., Hearing Components, Inc. v. Shure Inc.*, 600 F.3d 1357, 1366 (Fed. Cir. 2010) (reversing the district court's judgment of invalidity for indefiniteness); *Young v. Lumenis, Inc.*, 492 F.3d 1336, 1345 (Fed. Cir. 2007) (reversing the district court's judgment of invalidity for indefiniteness). Thus, absent a clear disclaimer of the scope of the claims, the disputed claim terms should be given their common, ordinary meanings.

AppSense currently asserts that three terms of the '170 patent are indefinite. The three terms currently in dispute are not indefinite and AppSense's arguments should be rejected.

1.   Personality Information Of The [Provided] Storage Medium:[16] "Personality Information Of The Source That Is Stored On The Storage Medium."

Tranxition's Construction	AppSense's Construction
Personality information of the source that is stored on the storage medium	Invalid as indefinite under 35 U.S.C. § 112(2)

The meaning of this term would not be insolubly ambiguous to one having ordinary skill in the art. One having ordinary skill in the art would understand that this term refers to the personality information of the source stored on the storage medium. Claims 1, 11 and 12 each require "providing a storage medium containing personality information of the source." Claims

---

[16] This term can be found in asserted claims 1, 11 and 12 of the '170 patent.

1, 11 and 12 further require, after each interception, "retrieving personality information of *the provided* storage medium." In light of the reference to "the provided" storage medium, a person of ordinary skill in the art understands that the retrieved personality information is the personality information contained on the storage medium as referenced earlier in the claim. The same is true of the personality information that is used to reply to the request in claims 1, 11 and 12. The specification, like the claims, discloses that the personality information on the storage medium is the personality information of the source. *E.g.*, '170 patent at 3:24-27, 4:58-64. In light of the intrinsic evidence, a person of ordinary skill in the art would not find that this term is insolubly ambiguous and would instead understand the term to have the meaning proposed by Tranxition. Tranxition's proposed construction should therefore be adopted.

Tranxition expects that AppSense will argue this term is indefinite because it refers to the personality *of* the storage medium. For the reasons stated above, a person of ordinary skill in the art would understand what this terms refer to. AppSense cannot meet its burden of showing by clear and convincing evidence that this term is insolubly ambiguous. AppSense's argument should therefore be rejected.

2.   Injecting Code Into The Process/Injects Code Into A Process:[17] "Insert[ing][s] Code To Be Executed Amid The Execution Of The Application Code."

Tranxition's Construction	AppSense's Construction
Insert[ing][s] code to be executed amid the execution of the application code	Invalid as indefinite under 35 U.S.C. § 112(2)

AppSense cannot show by clear and convincing evidence that this term is insolubly ambiguous. Indeed, the claim language itself is not ambiguous and refers to the well-known concept of inserting code (such as a snippet of an application or a dynamic link library) into a process. This plain meaning of the claim language is confirmed by both Dr. Arthur T. Brody, Tranxition's expert, and Dr. Samrat Bhattacharjee, AppSense's expert. Weiss Decl. Ex. B at ¶

---

[17] These terms can be found in asserted claims 6 and 20 of the '170 patent.

17; Weiss Decl. Ex. C at ¶ 17 ("The patent specification describes various method by which the startup interceptor DLL can modify the process's code, possibly injecting new code into the process"); ¶ 18 ("Execution of the startup interceptor DLL modifies the code of the process."). It is also undisputed that the specification describes injecting code into a process. *E.g.*, '170 patent at Figs. 3, 6, 4:44-58, 6:42-56, 7:28-45; Weiss Decl. Ex. B at ¶¶ 22-23; Weiss Decl. Ex. C at ¶¶ 17-18. Because there appears to be no dispute that the claim language is clear, these terms are not indefinite.

In spite of the undisputed understanding of the disputed term, AppSense, relying on Dr. Bhattacharjee, argues that these terms are indefinite because the scope of the terms, which are found in dependent claims, is the same as the scope of the independent claims. *E.g.*, Weiss Decl. Ex. D at ¶ 13. Even if the scope of the claims is same as opined by Dr. Bhattacharjee, it remains unclear how this opinion renders these terms indefinite. Tranxition is unaware of any legal support, and AppSense has not provided any legal support, for Dr. Bhattacharjee's opinion.

AppSense appears to alternatively argue that, if the scope of these terms is not the same as the independent claims, a person of ordinary skill in the art would not understand the scope of the terms. Weiss Decl. Ex. C at ¶ 13. AppSense, through Dr. Bhattacharjee, has offered no explanation for this assertion. As discussed above, the specification describes multiple embodiments where code is injected into a process and a person of ordinary skill in the art would not be confused as to the scope of the limitation. Instead, it appears that AppSense is arguing that the injected code must be a startup interceptor DLL. *See* Weiss Decl. Ex. D at ¶ 11 (Dr. Bhattacharjee dismissing an embodiment of injecting code because it was not related to a startup interceptor DLL). The injecting code terms do not, however, require that the injected code be related to the startup interceptor DLL. Claim 20, where the term "injects code into a process" is found, claims a new component that can be (although need not necessarily be) separate from the components claimed in claim 15, which include the startup interceptor DLL component. Similarly, in claim 1, the "injecting code" limitation is not written such that it is a further

21

limitation of the startup interceptor DLL. Neither AppSense nor Dr. Bhattacharjee has offered any clear and unambiguous disclaimers importing the limitation that it must be the startup interceptor DLL performing the injecting. Therefore, AppSense's alternative argument must also be rejected.

Accordingly, these terms are clear and are not indefinite. If the term needs a construction, the Court should adopt the following construction: insert[ing][s] code to be executed amid the execution of the application code.

3. Modifying Code Of The Process:[18] "Inserting Code To Execute In Place Of Application Code"

Tranxition's Construction	AppSense's Construction
Inserting code to execute in place of application code	Invalid as indefinite under 35 U.S.C. § 112(2)

The modifying term is not indefinite. Within the context of the claim language, a person of ordinary skill in the art would understand that this term refers to changing or altering the code a process (a process is a running instance of an application). *See* Weiss Decl. Ex. B at ¶ 28. The specification also uses this term according to its ordinary meaning. '170 patent at 3:9-19, 4:38-48, 7:49-8:3; Weiss Decl. Ex. B at ¶ 31. Indeed, it appears that AppSense agrees because AppSense appears to argue that it is only the use of this term in claim 13, not claim 24, that is indefinite. This term is therefore not ambiguous and not indefinite.

AppSense, through Dr. Bhattacharjee, appears to argue that the modifying term as used in claim 13 is insolubly ambiguous because independent claim 12 refers to modifying information of the application and this term (in dependent claim 13) refers to modifying code of the process. Weiss Decl. Ex. D at ¶¶ 15-17. The difference between application and process in claims 12 and 13 do not create an ambiguity; instead, claim 13 is a narrower, more specific limitation than the limitation in claim 12. Consistent with the plain meaning of the terms, Dr. Bhattacharjee agrees

---

[18] This term can be found in asserted claims 13 and 24 of the '170 patent.

that an application is a program stored on a device, whether it is running or not, and a process is an instance of an application that is running. Weiss Decl. Ex. C at ¶ 21. A person of ordinary skill in the art would therefore understand modifying the code of a process is a reference to modifying the computer code of the running application, rather than the broader, more general modification of information of the application. A person of ordinary skill in the art would not find this term ambiguous. *See* Weiss Decl. Ex. B at ¶¶ 28-33, Ex. E at ¶¶ 15-18.

Accordingly, these are not indefinite and AppSense's argument should be rejected. If the term needs a construction, the Court should adopt the following construction: inserting code to execute in place of application code.

M.   The Component Terms of Claims 15, 22 And 23 Are Not Means-Plus-Function Claims And Need No Construction.

	Tranxition's Construction	AppSense's Construction
a component that intercepts requests to retrieve personality information of the target	Plain and ordinary meaning	This term requires construction under 35 USC 112 ¶ 6. **Function**: intercepting requests to retrieve personality information of the target **Corresponding Structure**: A startup interceptor DLL as disclosed at Col.7, ln. 43 – Col. 8, ln. 3 and Fig. 7 and equivalents thereof.
a component that, when a request is intercepted, retrieves personality information of a source and replies to the request with the retrieved personality information of the source, rather than the personality information of the target	Plain and ordinary meaning	This term requires construction under 35 USC 112 ¶ 6. **Function**: when a request is intercepted, retrieving personality information of a source and replying to the request with the retrieved personality information of the source, rather than the personality information of the target. **Corresponding Structure**: Interceptor component as described at col. 3, ll.56-60;

		col. 4, ll. 41-53; col. 5, ll. 29-42; col. 6, ln. 42-col. 7 ln. 27, and Figs 1 and 3-5 and equivalents thereof.

AppSense argues that, although these terms are not written in classic means-plus-function (with the "means" language), these terms are nonetheless means-plus-function terms subject to 35 U.S.C. 112, ¶6.   Because these terms do not use the "means" term, there is a "strong [presumption] that is not readily overcome" that 35 U.S.C. § 112, ¶6 does not apply. *Lighting World, Inc. v. Birchwood Lighting, Inc.*, 382 F.3d 1354, 1358 (Fed. Cir. 2004).   Only by presenting evidence that the claim terms "fail[] to recite sufficiently definite structure or else recite[] function without reciting sufficient structure for performing that function" can AppSense rebut this strong presumption. *Id.* (internal quotations omitted).

AppSense cannot meet its burden of showing that these terms do not recite sufficient structure.   Tranxition believes that AppSense's argument is that component does not provide sufficient structure.   This argument has, however, been addressed and rejected by multiple courts. *E.g.*, *Widevine Techs., Inc. v. Verimatrix, Inc.*, Civil Action No. 2-07-cv-321, 2009 U.S. Dist. LEXIS 102768, at *45-46 (E.D. Tex. Nov. 4, 2009) (holding that component is sufficient structure where it "refers to software that is an element of the 'computer-readable storage medium' of claim 9 or an element of the 'encryption bridge' of claim 1"); *see also Corelogic Info. Solutions, Inc. v. Fiserv, Inc.*, Case No. 2:10-CV-132-RSP, 2012 U.S. Dist. LEXIS 135386, at *23-26 (E.D. Tex. Sep. 21, 2012) (finding "model development component" contains sufficient structure in light of the detailed specification); *Leader Techs., Inc. v. Facebook, Inc.*, 692 F. Supp. 2d 425, 432-33 (D. Del. 2010) (finding that component in the claim in question were not subject to means-plus-function doctrine); *ROY-G-BIV Corp. v. Fanuc Ltd.*, Civil Action No. 2:07-cv-418, 2009 U.S. Dist. LEXIS 127428, at *79-83 (E.D. Tex. Aug. 25, 2009) (finding "motion control component" to have sufficient structure in light of the specificity in the specification); *Visto Corp. v. Research In Motion, Ltd.*, 623 F. Supp. 2d 756, (E.D. Tex. 2008)

(finding "redirector component" to have sufficient structure in light of the claim limitation and presumption).

Like *Wildvine* and similar cases, these terms have sufficient structure and should not be found to be subject to 35 U.S.C. § 112, ¶6. The claims specifically teach that these components are components of a target computer. *E.g.*, '170 patent at claim 15. Claim 15 also expressly provides additional detail regarding the "component that intercepts requests" by requiring that the "component that intercepts" to be a startup interceptor DLL. Additionally, the majority of the specification is dedicated to discussing the components that intercept requests and that retrieve personality information. *E.g.*, '170 patent at Abstract, Figs. 1, 5, 6, 7, 2:31-50, 3:3-66, 4:36-5:8, 7:6-8:53. The intrinsic evidence therefore provides sufficient structure for the component claim language: a software program to be used on a target computer.[19]

The Court should not find that these terms are means plus function terms subject to 35 U.S.C. § 112, ¶6.

V.   CONCLUSION.

For the foregoing reasons, the Court should adopt Tranxition's proposed constructions that are consistent with the plain and ordinary meaning of the disputed terms in light of the intrinsic evidence. AppSense's proposed constructions should be rejected because they are primarily designed to import limitations and are inconsistent with well-accepted claim construction principals.

---

[19] AppSense's position is also inconsistent. Other asserted terms use the "component" language. *E.g.*, '170 patent at claims 19 and 20. AppSense does not assert that these terms should be subject to means-plus-function doctrine.

Dated this 5th day of July, 2013.

<div style="margin-left: 45%;">

Respectfully submitted,

RUSS AUGUST & KABAT

By: /s/ Andrew D. Weiss
    Andrew D. Weiss

Marc A. Fenster (pro hac vice)
Andrew D. Weiss (pro hac vice)

MANSFIELDLAW
Johnathan E. Mansfield, OSB No. 055390
Telephone 971.271.8615

Attorneys for Plaintiff
Tranxition, Inc.

</div>